

PATENT ABSTRACTS OF JAPAN

(11)Publication number : **2002-358136**

(43) Date of publication of application : 13.12.2002

(51)Int.Cl.

G06F 1/00

G06F 17/60

(21)Application number : 2002-077890

(71)Applicant : INTERNATL BUSINESS MACH
CORP <IBM>

(22)Date of filing : 20.03.2002

(72)Inventor : CARBONE GIANCARLO
CIERECH THOMAS F

(30)Priority

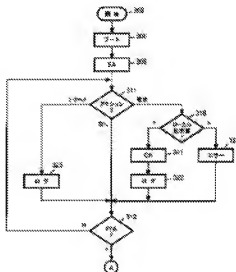
Priority number : 2001 200107321 Priority date : 23.03.2001 Priority country : GB

(54) METHOD AND SYSTEM FOR CONTROLLING USE OF SOFTWARE PROGRAM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a method and system for controlling the use of a software program on a client.

SOLUTION: This program (operating system or network stack) is started before a boot strap process is completed. Each program requests execution permission to a license issue management system (311), and a system agent applies the local license permission of each program on a client before the boot strap is completed (318–324). The all function agent of the license issue management system revises the applied permission according to the verification of a license certificate stored on a server work station before the boot strap is completed (342).



(51) Int. Cl. ⁷	識別記号	F I	チーフド (参考)
G 0 6 F 1/00		G 0 6 F 17/60	1 3 2 5 B 0 7 6
17/60	1 3 2		1 4 2
	1 4 2	9/06	6 6 0 C

審査請求 有 請求項の数14 O L (全 11 頁)

(21) 出願番号 特願2002-77890 (P2002-77890)

(22) 出願日 平成14年3月20日 (2002. 3. 20)

(31) 優先権主張番号 0 1 0 7 3 2 1 . 2

(32) 優先日 平成13年3月23日 (2001. 3. 23)

(33) 優先権主張国 イギリス (G B)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション
INTERNATIONAL BUSIN
ESS MASCHINES CORPO
RATION
アメリカ合衆国10504、ニューヨーク州
アームロン ニュー オーチャード ロー
ド

(74) 代理人 100086243

弁理士 坂口 博 (外1名)

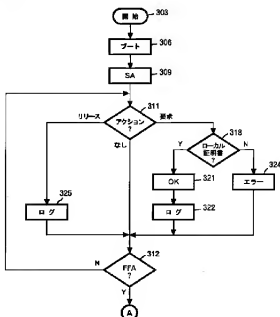
最終頁に続く

(54) 【発明の名称】 ソフトウェア・プログラムの使用を制御するための方法およびシステム

(57) 【要約】 (修正有)

【課題】 クライアント上のソフトウェア・プログラムの使用を制御する方法およびシステム。

【解決手段】 プログラム (オペレーティング・システムまたはネットワーク・スタック) は、ブートストラップ・プロセスの完了前に開始する。各プログラムは、ライセンス交付管理システムに対して実行許可を要求し (311)、システム・エージェントは、ブートストラップの完了前に、クライアント上の各プログラムについてのローカル・ライセンス許可を授与する (318~324)。ライセンス交付管理システムの全機能エージェントは、ブートストラップの完了前に、サーバ・ワークステーション上に格納されたライセンス証明書の検証に従って、授与された許可を改訂する (342)。



【特許請求の範囲】

【請求項1】コンピュータ上のソフトウェア・プログラムの使用を制御する方法（300）であって、前記コンピュータに対して基本サービスを提供する少なくとも1つのソフトウェア・プログラムをロードすることに関係するブートストラップを開始するステップ（306）と、ライセンス交付管理システムに対して各プログラムによって実行許可を要求するステップ（311）とを含み、前記ブートストラップの完了前に、前記プログラムの使用の許可条件を示すライセンス交付情報の仮検証に従って前記許可を授与するステップ（318～324）と、前記ブートストラップ完了後に、前記ライセンス交付情報の完全な検証に従って前記授与した許可を改訂するステップ（342～351）とを特徴とする方法。

【請求項2】前記ブートストラップの完了前に、前記ライセンス交付管理システムの基本モジュールをロードするステップ（309）であって、各許可が、前記基本モジュールに対して要求され、前記基本モジュールによって授与されるステップと、

許可の各要求を記録するステップ（322）と、前記ブートストラップ完了後に、前記ライセンス交付管理システムの全モジュールをロードするステップ（312）と、前記対応する授与された許可を改訂するために、記録した各要求を前記全モジュールに供給するステップ（342）とをさらに含む請求項1に記載の方法（300）。

【請求項3】前記許可を要求するステップ（311）が、前記基本モジュールの前記ローディングが完了するまで延期される請求項2に記載の方法（300）。

【請求項4】前記授与された許可を改訂する前記ステップ（342～351）が、前記対応するライセンス情報を含む、各プログラムについてのデジタル・ライセンス証明書格納する別のコンピュータに、前記要求を示すメッセージを送信するステップ（343）と、前記対応するライセンス証明書で、前記要求の適合を検証するステップ（345～351）とを含む請求項2または3に記載の方法（300）。

【請求項5】前記許可を授与する前記ステップが、前記コンピュータ上に格納され、前記対応するライセンス証明書の簡略化したコピーを含むローカル・デジタル・ライセンス証明書を検証するステップ（318～324）を含む請求項4に記載の方法（300）。

【請求項6】前記ブートストラップの後に、前記別のコンピュータから受信するその新しいバージョンに従って各ローカル・ライセンス証明書を更新するステップ（354）をさらに含む請求項5に記載の方法（300）。

【請求項7】各プログラムの実行環境の変化を検出するために、前記コンピュータのオペレーションを監視する

ステップ（333）と、前記検出した変化に従って前記対応する授与された許可を改訂するステップ（336、360～384）とをさらに含む請求項1ないし6に記載の方法（300）。

【請求項8】コンピュータ上で動作するときに請求項1ないし7のいずれかの前記方法を実行するための、前記コンピュータ（105）の作業メモリ（130c）中に直接ロード可能なコンピュータ・プログラム（230、235）。

【請求項9】請求項8のプログラム（230、235）が格納されるコンピュータ可読記録媒体（150）。

【請求項10】コンピュータ上のソフトウェア・プログラムの使用を制御する方法（300）であって、ライセンス交付管理システムに対して、各プログラムによって実行許可を要求するステップ（311；327）と、

前記プログラムの使用の許可条件を示すライセンス交付情報の検証に従って前記許可を授与するステップ（318～321、342～351；330、357～372）とを含み、

前記コンピュータの実行環境での変化を検出するために、前記コンピュータのオペレーションを監視するステップ（333）と、

前記検出した変化に従って前記授与された許可を改訂するステップ（336、360～384）とを特徴とする方法。

【請求項11】コンピュータ上で動作するときに請求項10の前記方法を実行するための、前記コンピュータ（105）の作業メモリ（130c）中に直接ロード可能なコンピュータ・プログラム（230、235）。

【請求項12】請求項11のプログラム（230、235）が格納されるコンピュータ可読記録媒体（150）。

【請求項13】ライセンス交付管理システム（230、235）と、コンピュータ（105）に対して基本サービスを提供する少なくとも1つのソフトウェア・プログラム（205c、215c）のローディングに関係するブートストラップを開始するための手段（135）とを含む前記コンピュータ上のソフトウェア・プログラムの使用を制御するためのシステム（100）において、各プログラムが、前記ライセンス交付管理システムに対して実行許可を要求するシステムであって、前記ライセンス交付システムが、前記ブートストラップの完了前に、プログラムの使用の許可条件を示すライセンス交付情報（240）の仮検証に従って前記許可を授与するための手段（230）と、前記ブートストラップの完了後に、ライセンス交付情報（225）の完全な検証に従って前記授与した許可を改訂するための手段（235）とを含むことを特徴とするシステム。

【請求項14】ライセンス交付管理システム（230、

3

235)を含むコンピュータ(105)上のソフトウェア・プログラム(205、215、250)の使用を制御するシステム(100)において、各プログラムが、前記ライセンス交付管理システムに対して実行許可を要求し、前記ライセンス交付管理システムが、ライセンス交付情報(225、240)の検証に従って許可を授与するシステムであって、前記プログラムの実行環境の変化を検出するために、コンピュータのオペレーションを監視するための手段(230、233)をさらに含み、前記ライセンス交付管理システムが、前記検出した10 変化に従って前記授与した許可を改訂することを特徴とするシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ソフトウェア・プログラムの使用を制御するための方法およびシステムに関する。

【0002】

【従来の技術】 コンピュータ上で動作するソフトウェア・プログラムは、無数の数のコピーに完全に複製することができ、これは、自身の知的所有権を保護することを望む、プログラムの発表者にとって主要な関心事である。実際、発表者は、一般に各プログラムの使用許諾に対して使用料を受け取るので、プログラムのどんな不明な使用または配布によっても、使用料が支払われなくなる。この問題は、この種の製品の配布を規制なしに配布することをさらに容易にするインターネットの広範にわたる拡散によって、近年悪化してきている。

【0003】 プログラムの不明な使用を回避する最も直接的な方法は、無許可のコピーおよび伝送を防止するものである。例えば、大部分のプログラムは、作成することができるコピーの数を制限し、または所定の期間が経過した後にプログラムのオペレーションを使用不能にする制御コードを組み込む。別の技法は、プログラムの実行に対して、ソフトウェアの所有権またはハードウェアを要求することからなる。

【0004】 当技術分野で提案された異なる解決策は、発表者によって許可されたそれぞれの使用の条件(例えば、プログラムが動作することのできるコンピュータの最大計算能力を定義する条件)で、コンピュータ上で動作するプログラムの適合を制御するライセンス交付管理システムを、コンピュータ上にインストールするものである。より具体的に、(マシン可読形式に変換された)各プログラムについての使用の許可条件は、デジタル・ライセンス証明書中に組み込まれる。このプログラムは、ライセンス交付管理システムに対して呼び出しを含み、その結果、エンドユーザがプログラムの実行を要求する度に、対応する要求がライセンス交付管理システムに伝送される。プログラムの実行が、ライセンス証明書中に組み込まれた使用の許可条件によって設定された

4

制限内にあるかどうかを、このライセンス交付管理システムは検証する。このライセンス交付管理システムは、この検証の結果に従ってプログラムの実行を動作可能にし、または防止する。

【0005】

【発明が解決しようとする課題】 しかし、上述の解決策は、完全に満足のゆくものではない。具体的には、ライセンス交付管理システムの実行は、オペレーティング・システムと、コンピュータに対して基本サービスを提供する他のプログラムとの全機能を必要とする。したがって、コンピュータの電源を投入する度に、基本プログラムがその間にロードされるブートストラップ・プロセスが最初に実行される。ブートストラップが完了すると、他のプログラムがその上で動作することができるソフトウェア・プラットフォームが提供され、ライセンス交付管理システムを開始することができる。

【0006】 その結果、ブートストラップ・プロセスの間にロードされる(オペレーティング・システム自体、ネットワーク・スタックなどの)すべての基本プログラムは、ライセンス交付管理システムで監視することができない。これにより、コンピュータ上で動作するプログラムの全制御を実施することが妨げられる。上述の欠点は、制御から逃れるプログラムが一般に非常に価値のあるものであるので、特に重大である。したがって、プログラムの発表者が受ける経済的損失は、かなりのものとなる可能性がある。

【0007】 上述の欠点を克服することが本発明の目的である。この目的を達成するために、請求項1および10に記載の方法を提案する。

【0008】

【課題を解決するための手段】 簡潔に言えば、本発明は、コンピュータ上のソフトウェア・プログラムを制御する方法であって、前記コンピュータに対して基本サービスを提供する少なくとも1つのソフトウェア・プログラムをロードすることに関係するブートストラップを開始するステップと、ライセンス交付管理システムに対して各プログラムによって実行許可を要求するステップとを含む方法を提供する。この方法は、前記ブートストラップの完了前に、前記プログラムの使用の許可条件を示すライセンス交付情報の仮検証に従って前記許可を授与するステップと、前記ブートストラップ完了後に、前記ライセンス交付情報の完全な検証に従って前記授与した許可を改訂するステップとをさらに含む。

【0009】 あるいは、本発明は、コンピュータ上のソフトウェア・プログラムの使用を制御する方法であって、ライセンス交付管理システムに対して、各プログラムによって実行許可を要求するステップと、前記プログラムの使用の許可条件を示すライセンス交付情報の検証に従って前記許可を授与するステップとを含む方法を提供する。この方法は、前記コンピュータの実行環境での

変化を検出するために、前記コンピュータのオペレーションを監視するステップと、前記検出した変化に従って前記授与された許可を改訂するステップとをさらに含む。

【0010】さらに、本発明は、これらの方法を実行するためのコンピュータ・プログラムと、プログラムを格納する製品と、ソフトウェア・プログラムの使用を制御するための対応するシステムも提供する。

【0011】

【発明の実施の形態】具体的に図1を参照すると、LAN (ローカル・エリア・ネットワーク) などのネットワーク・システム100が示されている。LAN100は、複数のワークステーション105によって形成される場合、一般にPC (パーソナル・コンピュータ) からの。ワークステーション105は、それぞれの通信ケーブル115を介して、1つまたは複数の集線装置110 (ハブなど) に接続される。ネットワーク・システム100は、ワークステーション105のうちの (一般に強力なコンピュータからなる) 1つまたは複数が共有ファイルおよび共有装置などのネットワーク・リソースを管理するクライアント/サーバ・アーキテクチャを有する。他のワークステーション105は、ユーザがサーバ・リソースに依拠してアプリケーションをその上で実行するクライアントとして動作する。

【0012】各ワークステーション105は、通信バス120に並列に接続されるいくつかのユニットを含む。具体的には、中央演算処理装置 (CPU) 125は、ワークステーション105のオペレーションを制御し、作業メモリ130 (一般にはDRAM) は、CPU125によって直接的に使用され、読み取り専用メモリ (ROM) 135は、ワークステーション105を開始するための基本プログラムを格納する。様々な周辺装置が、(それぞれのインターフェースによって) バス120にさらに接続される。具体的には、大容量メモリは、ハードディスク140と、CD-ROM (コンピュータ読み取り可能な記録媒体) 150を読み取るためのドライバ・ユニット (DRV) 145とからなる。ワークステーション105は、例えばキーボードおよびマウスからなる入力ユニット (I/U) 155と、例えばモニタからなる出力ユニット (O/U) 160とをさらに含む。ワークステーション105は、ネットワーク・インターフェース・カード (NIC) 165を使用して (通信ケーブル115を介して) 集線装置110に接続される。

【0013】このネットワーク・システムがWAN (ワイド・エリア・ネットワーク) からなる場合、このネットワーク・システムが (1つまたは複数のスイッチなどの) 異なる構成要素を含む場合、各ワークステーションが (例えばマルチプロセッサ・アーキテクチャを有する) 異なる構造を有する場合、このネットワーク・システムが、一連の端末を備えるメインフレーム

か、あるいは1つまたは複数のコンピュータを備える他のデータ処理システムによって置換される場合にも同様の考慮事項が当てはまる。

【0014】次に図2を考慮すると、クライアントとして動作するワークステーションの作業メモリ (130cと示す) の部分的内容と、ライセンス交付サーバとして動作するワークステーションの作業メモリ (130sと示す) の部分的内容とが示されている。情報 (プログラムおよびデータ) は、一般に、ハードディスク上に格納され、プログラムが実行されるときに作業メモリ130c、130s中にロードされる。このプログラムは、最初に、CD-ROMから、クライアント・ワークステーションのハードディスク上と、サーバ・ワークステーションのハードディスク上とにインストールされる。

【0015】オペレーティング・システム (OS) 205cおよびオペレーティング・システム205sは、それぞれ、他のプログラムがその上で動作することのできるクライアント・ワークステーションおよびサーバ・ワークステーションに対してソフトウェア・プラットフォームを提供する。オペレーティング・システム205cおよび205sは、キーボードからの入力を認識すること、モニタに出力を送信すること、およびハードディスク上のファイルを追跡することなどの基本タスクを実行する。各オペレーティング・システム205c、205sのメイン・モジュール (カーネルとして知られる) は、それぞれ作業メモリ130c、130s中に常駐する。カーネルは、オペレーティング・システム205c/205sの他の部分が必須とする、不可欠なサービスすべて (プログラムを作業メモリ中にロードすること、プロセスをスケジューリングすること、またはコマンドを解釈することなど) を提供する。スタック (STACK) 215cおよびスタック215sは、ネットワーク・システムを介して通信を定義するために共に働く、1組のネットワーク・プロトコル層を処理する。具体的には、スタック215c、215sにより、クライアント・ワークステーションおよびサーバ・ワークステーションが両者の間でメッセージを交換することが可能となる。

【0016】ライセンス交付管理アプリケーション (LMA) 220は、サーバ・ワークステーション上に格納されたデジタル・ライセンス証明書 (CERT) のデータベースを制御する。ネットワーク・システム上で動作する各ソフトウェア・プログラムの使用を許可するために、異なるライセンス証明書225が使用される。ライセンス証明書225は、証明書の識別子、プログラムの発表者の識別子、発表者によって許可されたときのプログラムの使用の条件、およびプログラムがその下でまだ動作することのできる (許可された条件の外の) 使用の例外条件を含む。

【0017】例えば、プログラムの使用の許可条件は、

プログラムが動作することのできるワークステーションのCPUの最大能力または作業メモリの最大容量、プログラムを同時に実行することができるエンドユーザの最大数などを定義する。さらに、発表者は、使用の実際の条件が許可された条件を事前設定した場合（10%など）だけ超過したときでも、プログラムの動作することを例外的に可能にする。例えば100MIPSに等しいCPUの最大能力でワークステーション上で動作するようにプログラムが許可される場合、CPUの能力が最大110MIPSまで増加した場合でも、（使用の例外条件の下で）プログラムはまだ動作することができる。

【0018】各ライセンス証明書225は、発表者によって作成され、対応する秘密キーを使用して署名される。発表者に関連するデジタル・シグニチャおよびデジタル証明書は、ライセンス証明書225に附加される。次いで、ライセンス交付管理アプリケーション220は、ライセンス証明書225を妥当性検査するために、対応するデジタル証明書から発表者の公開キーを抽出し、デジタル・シグニチャを検証することができる。

【0019】（経路）ライセンス交付システム・エージェント（SA）230は、クライアント・ワークステーション上のオペレーティング・システム205cのカーネルの一部である。システム・エージェント230により、ライセンス交付管理システムの（非常に小さく、明確なタスクからなる）基本オペレーションが提供される。システム・エージェント230は、クライアント・ワークステーションの構成に関する情報（CPUの能力、作業メモリの容量など）を格納する、カーネル内部のデータ構造（CONF）233にアクセスする。構成情報233は、クライアント・ワークステーション上で動作するプログラムの実行環境を定義する。

【0020】全機能ライセンス交付エージェント（FFA）235は、クライアント・ワークステーション上で一層アクティブなデモン・プロセスであり、ライセンス交付管理システムの完全なオペレーションを提供する。全機能エージェント235は、（サーバ・ワークステーションとメッセージを交換するために）システム・エージェント230およびスタック215cと通信する。

【0021】システム・エージェント235は、ローカル・デジタル・ライセンス証明書（LOCALCERT）240のリポジトリを制御する。各ローカル・ライセンス証明書240は、（サーバ・ワークステーション上に格納された）対応するライセンス証明書225の簡略化したコピーを含む。具体的には、ローカル・ライセンス証明書240は、単に、クライアント・ワークステーション上で動作することを許可された対応するプログラムの識別子からなる。システム・エージェント230は、さらに、基本プログラム（オペレーティング・システム205cおよびスタック215c）から受信した、許可

またはリリースの要求を格納するログ構造（LOG）245を制御する。オペレーティング・システム205c上で動作するアプリケーション・プログラム250は、さらに、許可またはリリースのそれぞれの要求を送信するために、システム・エージェント230と通信する。

【0022】プログラムおよびデータが異なる方式で構築される場合、他のモジュールまたは機能が提供される場合、異なる使用の条件が考えられる場合、プログラムが使用の許可条件の外で動作することができない場合、クライアント・ワークステーションについての基本サービスを提供する1つまたは複数の異なるプログラムが（オペレーティング・システム単体）提供される場合などに、同様の考慮事項が当てはまる。

【0023】図3～5に示されているように、共に方法300を構成する一連のルーチンが、クライアント・ワークステーション上にインストールされたプログラムの使用を制御するために、時間的に連続するステージで実行される。方法300は、ブロック303で開始し、次いでブロック306に進み、クライアント・ワークステーションがオンにされる。その結果、CPUは、（内部電源が安定化した後に）ROMメモリ中に格納された基本プログラムをロードする。基本プログラムは、インストールされたハードウェアのテストを実行し、クライアント・ワークステーションを初期化する。次いで基本プログラムは、オペレーティング・システムを開始するためのブート・コードを格納する周辺装置を識別する。このコードは基本プログラムを引き継ぎ、ブートストラップ・プロセスを開始する。ブートストラップは、クライアント・ワークステーションについての不可欠なサービスを提供するユーティリティをロードすることを含み、ユーティリティは、ブートストラップが進行するにつれて、オペレーティング・システムの残りのローディングおよび実行と、他の基本プログラム（ネットワーク・スタックなど）のローディングおよび実行とを制御する。

【0024】ブロック309に進むと、ライセンス交付管理システムのシステム・エージェントが、ブートストラップ中に（オペレーティング・システムのカーネルの一部として）ロードされる。ブロック311で、システム・エージェントは、（問題とする実施例中のオペレーティング・システムまたはネットワーク・スタックである）基本プログラムが開始したか、それともブートストラップ中に終了したかをチェックする。

【0025】システム・エージェントに対してアクションが通知されなかった場合、この方法は（以下で説明した）ブロック312に下る。逆に、基本的な問題が開始したとき、この基本的な問題は、（システム・エージェントが完了した後）システム・エージェントに実行許可を要求する。その結果、ブロック318で、システム・エージェントは、ローカル・ライセンス証明書のリポジトリを検証する。基本プログラムについてのローカル

・ライセンス証明書が見つかった（基本プログラムの実行が許可されていることを示す）場合、システム・エージェントは、ブロック321で基本プログラムの実行を可能にし、次いでブロック322で許可の要求を記録する。見つからなかった場合、基本プログラムの実行が打ち切られ、この方法は、ブロック324でエラー条件に入る（このエラー条件は、一般に後の分析のために記録される）。どちらの場合でも、この方法はブロック312に進む。基本プログラムがその実行を終了するとき

（ブロック311）、基本プログラムは、それに応じて（終了する前に）システム・エージェントに通知する。その結果、システム・エージェントは、ブロック325で対応するリリースの要求を記録し、次いでブロック312に下る。

【0026】次にブロック312を考慮すると、ブートストラップが終了した後、全機能エージェントがロードされる。全機能エージェントのローディングが完了するとすぐに、それに応じてシステム・エージェントが通知される。システム・エージェントが全機能エージェントのローディングをまだ通知していない場合、この方法は、上述のステップを反復するためにブロック311に戻る。システム・エージェントが全機能エージェントのローディングを通知していた場合、この方法は、並列に実行される2つの分岐に分岐する。第1分岐は、ブロック327～336からなり、第2分岐は、ブロック342～384からなる。この2つの分岐はブロック387（以下で説明する）で接合する。

【0027】次にブロック327を考慮すると、システム・エージェントは、アプリケーション・プログラムが開始または終了したかどうかをチェックする。開始または終了した場合、システム・エージェントは、ブロック330で、全機能エージェントに対して、対応する許可またはリリースの要求を転送し、次いでこの方法は、ブロック333に進む。開始または終了しなかった場合、この方法は、直接ブロック333に下る。次にブロック333を考慮すると、システム・エージェントは、クライアント・ワークステーション上で動作する（基本およびアプリケーション）プログラムの実行環境を定義する構成情報の変化が生じたかどうかを検証する。変化が生じた場合、ブロック336で新しい構成情報が全機能エージェントに供給され、次いで、この方法はブロック387に進む。変化が生じなかった場合、この方法は直接ブロック387に下る。

【0028】同時に、ブロック342で、ブートストラップの間に格納された基本プログラムからの許可またはリリースの要求が、システム・エージェントによってログ構造から抽出され、全機能エージェントに供給される。このようにして得られた要求は、対応する実行環境を定義する構成情報と共に、ブロック343で（それぞれのネットワーク・スタックを介して）サーバ・ワーク

ステーションに送信される。ブロック345に進むと、サーバ・ワークステーション上のライセンス交付管理アプリケーションが、それぞれのライセンス証明書中に含まれる使用の許可条件で、各要求の適合を検証する。基本プログラムにマッチするライセンス証明書が見つかり、かつ基本プログラムが発表者によって許可された使用の条件内で動作している場合（例えば、CPUの能力が可能な最大値を超えていない場合）、（システム・エージェントによって以前に与えられた）許可がブロック348で確認される。そうでない場合、この方法は、ブロック351でエラー条件に入る（このエラー条件は後の分析のために記録される）。どちらの場合でも、対応する応答メッセージが、全機能エージェントに送信される。一般に、応答メッセージは、許可が授与されたか、それとも拒否されたかを指定するリターン・コードを含み、後者の場合にはその拒否の理由を含む。さらに、ステータス・コードは、授与された許可が使用の（標準）条件内にあるか、それとも使用の例外条件内にあるかをさらに指定し、または拒否の理由をより詳細にさらに指定する。

【0029】次いでこの方法は、（ブロック348およびブロック351の両方から）ブロック354に入り、基本プログラムについてのローカル・ライセンス証明書の新しいバージョンが、サーバ・ワークステーション上に格納されたそれぞれのライセンス証明書から生成される。ローカル・ライセンス証明書の新しいバージョンは、ライセンス交付管理アプリケーションから、クライアント・ワークステーション上の全機能エージェントに送信される。全機能エージェントは、それに応じて、システム・エージェントにローカル・ライセンス証明書のリポトリを更新させる。

【0030】次いでこの方法はブロック357に下り、リリースの許可の要求がアプリケーション・プログラムによって（システム・エージェントを介して）要求されたかどうかを、全機能エージェントがチェックする。要求されなかった場合、この方法はブロック360（以下で説明する）に進む。要求された場合、この要求は、対応するライセンス交付情報を更新するために、ブロック363でサーバ・ワークステーションに送信される。ブロック366に進むと、アプリケーション・プログラムが要求した実行許可を有する場合、ライセンス交付管理アプリケーションは、それぞれのライセンス証明書に含まれる使用の許可条件で、要求の適合を検証する。検証の結果が肯定的である場合、許可がブロック369で授与される。検証の結果が否定的である場合、この方法はブロック372でエラー条件に入る。どちらの場合でも、対応する応答メッセージ（リターン・コードおよびステータス・コード）が全機能エージェントに送信される。この応答メッセージはシステム・エージェントに送信され、次いで許可を要求するアプリケーション・プロ

11

グラムに返され、それに応じてアプリケーション・プログラムはその実行を続行または打ち切る。

【0031】この方法は、(ブロック369およびブロック372の両方から)ブロック360に進み、新しい構成がシステム・エージェントから受信されたかどうかを、全機能エージェントが検証する。新しい構成情報が受信されなかった場合、この方法は直接ブロック387に下る。新しい構成情報が受信された場合、この新しい構成情報が、ブロック375でサーバ・ワークステーションに送信される。ブロック378に進むと、ライセンス交付管理アプリケーションは、新しい構成情報に照らして授与した各許可を改訂する。各(基本およびアプリケーション)プログラムの実行が、まだそれぞれのライセンス証明書に含まれる使用の許可条件を満たしている場合、この許可はブロック381で確認される。満たしていない場合、この方法はブロック384でエラー条件に入る。どちらの場合でも、対応する応答メッセージが全機能エージェントに送信され、この方法はブロック387に進む。

【0032】次にブロック387を考慮すると、クライアント・ワークステーションがシャットダウンされたかどうかのチェックが行われる。シャットダウンされていない場合、この方法は(上述のステップを反復するために)ブロック312の後次の分岐点に戻る。シャットダウンされた場合、この方法は最終ブロック390で終了する。

【0033】例えば(クライアント・ワークステーションがオフの位置からオンにされる)コールド・ブートストラップの代わりに、(クライアント・ワークステーションがリセットされる)ウォーム・ブートストラップを用いて等価な方法が実行される場合、エラーが発生したときにプログラムの実行が動作不能にされる場合、応答メッセージが異なる構造を有する場合などに、同様の考慮事項が当てはまる。

【0034】より一般的には、本発明は、コンピュータについての基本サービスを提供する1つまたは複数のソフトウェア・プログラムをロードすることに関係するブートストラップを開始するステップと、ライセンス交付管理システムに対して、各プログラムによって実行許可を要求するステップとを含む、コンピュータ上のソフトウェア・プログラムの使用を制御する方法を提供する。この方法は、ブートストラップの完了前に、プログラムの使用の許可条件を示すライセンス交付情報の仮検証に従って許可を授与し、ブートストラップの完了後に、ライセンス交付情報の完全な検証に従って授与した許可を改訂するステップをさらに含む。

【0035】本発明の解決策により、コンピュータ上で動作するプログラムの完全な制御を実装することが可能となる。

【0036】具体的には、提案の方法により、ブートス

12

トラップ・プロセスの間にロードされるプログラム(オペレーティング・システムおよびネットワーク・スタックなど)の(ライセンス交付管理システムによる)効果的な監視が可能となる。実際、上述の解決策では、プログラムの使用は、オペレーティング・システムの全機能が利用可能となる前でも制御することができ。

【0037】本発明による解決策により、発表者は自身の知的所有権を効果的に保護することが可能となる。実際、基本プログラムおよびアプリケーション・プログラムの両方の、どんな不明な使用も防止され、使用料の未払いによるどんな経済的損失も回避される。あるいは、このように収集された情報は、顧客によって行われるプログラムの実際の使用を記録するための、発表者によって利用される。例えば、マーケティングまたは統計の応用のために、この情報を有利に実施することができ。

【0038】上述の本発明の好ましい実施形態は、別の利点も提供する。例えば、システム・エージェントおよび(対応するログ構造を有する)全機能エージェントの提供により、非常に単純にライセンス交付管理システムが実装される。このように、本発明のプロセスは、それぞれの実行許可をライセンス交付管理システムに要求する(基本およびアプリケーション)プログラムに対して完全に透過である。さらに、軽量エージェントのローディングの完了まで許可の要求を延期する機能により、ブートストラップの間の非常に初期に開始するプログラムも監視することが可能となる。具体的には、この解決策により、オペレーティング・システムのカーネル自体の使用を制御することが可能となる。

【0039】ライセンス交付管理システムが等価な基本モジュールおよび全モジュールに区分される場合、(基本プログラムからの)格納された許可またはリリースの要求が全機能エージェントに対して異なる方式で提供される場合(例えば、ログ構造から直接抽出される場合)、許可の要求だけが記録される場合(例えば、ブートストラップの完了前には基本プログラムを終了することができない場合)、後に許可が基本プログラムからシステム・エージェントに対して要求される場合などに、同様の考慮事項が当てはまる。しかし、本発明の解決策は、(ライセンス証明書の仮検証および完全検証のどちらも実行する)単一モジュールからなるライセンス交付管理システムの場合でも、または後の検証のために、システム・エージェントのローディングの前に要求される各許可を記録することによっても実装される。

【0040】本発明の解決策は、クライアント・ワークステーション上で動作するプログラムについてのライセンス証明書がサーバ・ワークステーション上に格納されるネットワーク・システムで特に有利である。このように、ネットワーク・スタックによって提供される通信機能が利用可能となる前であっても、プログラムの使

用を制御することができる。好ましくは、この許可は、クライアント・ワークステーション上に格納される。それぞれのローカル・ライセンス証明書の検証に従って授与される。この機能により、要求の検証が特に単純かつ高速になる(かつ、次いでブートストロップ中の実行に適するようになる)。さらに、ローカル・ライセンス証明書は、各ブートストロップの後に、(サーバ・ワークステーションから受信する)それぞれの新しいバージョンで継続的に置換される。それにより、クライアント・ワークステーション上に格納されたローカル・ライセンス証明書が、可能な範囲で常に最新となる。

【0041】クライアント・ワークステーションおよびサーバ・ワークステーションが異なる方式で通信する場合、異なるタイプのコンピュータを使用する場合、ライセンス証明書およびローカル・ライセンス証明書が異なる情報を含む場合(例えば有効期限)、ローカル・ライセンス証明書が異なる頻度で更新される場合(例えば2回または3回のブートストロップごとに、あるいは周期的に)などに、同様の考慮事項が当てはまる。

【0042】あるいは、ライセンス証明書およびローカル・ライセンス証明書が、プログラムの使用の許可条件を示す等価なライセンス交付情報によって置換され、許可が、ライセンス交付情報の異なる仮検証に従って授与され、ライセンス交付情報の異なる完全な検証に従って改訂される。しかし、本発明の方法は、ローカル・ライセンス証明書を更新することなく、かつローカル・ライセンス証明書なしに(例えば、ブートストロップの完了前にコンピュータ上のライセンス証明書の存在を単にチェックし、ブートストロップの完了後に許可条件を検証することによって)単一コンピュータ中でも使用される。

【0043】好ましくは、プログラムの実行環境のどんな変化も検出する目的で、プログラムが動作しているクライアント・ワークステーションのオペレーションが監視される。この情報が使用されて、それに応じて授与された許可が改訂される。このように、プログラムの使用を動的に追跡することが可能となる。例えば、クライアント・ワークステーションにより、オペレーションの間にCPUの能力を変更することができる場合、考案の解決策により、ライセンス交付情報をプログラムの使用の実際の条件と常に完全に整合させることが可能となる。

【0044】異なる構成パラメータを考慮する場合、クライアント・ワークステーションのオペレーションが異なる方式で(例えば専用の別のエージェントで)監視される場合、授与された許可が異なる頻度で更新される場合(例えば周期的に)などに、同様の考慮事項が当てはまる。しかし、本発明の解決策は、プログラムの実行環境を監視しない場合でも実装される。

【0045】有利には、本発明によって提案される方法は、CD-ROM上で提供されるコンピュータ・プロ

ラムで実装される。

【0046】あるいは、このプログラムは、フロッピー(R)・ディスク上で提供され、ハードディスク上に事前ロードされ、または他のコンピュータ可読媒体上に格納され、ネットワークを介してコンピュータに送信され、同報通信され、あるいはより一般には、コンピュータの作業メモリ中に直接ロード可能な任意の他の形態で提供される。しかし、本発明による方法は、例えば半導体材料のチップ中に集積された、クライアント・ワークステーション上に設置されたハードウェア構造でも実施される。

【0047】上述のライセンス交付情報の仮検証または完全検証を用いない場合であっても、コンピュータの監視オペレーションの追加の機能は、(単独の、または他の追加の機能と組み合わせた)使用に適することを留意されたい。

【0048】より一般には、本発明は、ライセンス交付管理システムに対して各プログラムによって実行許可を要求するステップと、プログラムの使用の許可条件を示すライセンス交付情報の仮検証に従って許可を授与するステップとを含む、コンピュータ上のソフトウェア・プログラムの使用を制御する方法をさらに提供する。この方法は、プログラムの実行環境の変化を検出するために、コンピュータのオペレーションを監視するステップと、その検出した変化に従って授与された許可を改訂するステップとをさらに含む。

【0049】例えば、基本プログラム(オペレーティング・システムなど)の使用を制御することを許可しないライセンス交付管理システムでは、この機能を利用して、アプリケーション・プログラムの使用を動的に追跡することが有利である。

【0050】もちろん、局所要件および特定の要件を満たすために、当業者は上述の解決策を多くの修正形態および変更形態に適用することができるが、そのすべては、前記の特許請求の範囲で定義される本発明の保護の範囲内に含まれている。

【0051】まとめとして、本発明の構成に関して以下の事項を開示する。

【0052】(1)コンピュータ上のソフトウェア・プログラムの使用を制御する方法(300)であって、前記コンピュータに対して基本サービスを提供する少なくとも1つのソフトウェア・プログラムをロードすることに関係するブートストロップを開始するステップ(306)と、ライセンス交付管理システムに対して各プログラムによって実行許可を要求するステップ(311)とを含み、前記ブートストロップの完了前に、前記プログラムの使用の許可条件を示すライセンス交付情報の仮検証に従って前記許可を授与するステップ(318~324)と、前記ブートストロップ完了後に、前記ライセンス交付情報の完全な検証に従って前記授与した許可を改

訂するステップ(342~351)とを特徴とする方法。

(2) 前記ブートストラップの完了前に、前記ライセンス交付管理システムの基本モジュールをロードするステップ(309)であって、各許可が、前記基本モジュールに対して要求され、前記基本モジュールによって授与されるステップと、許可の各要求を記録するステップ(322)と、前記ブートストラップ完了後に、前記ライセンス交付管理システムの全モジュールをロードするステップ(312)と、前記対応する授与された許可を改訂するために、記録した各要求を前記全モジュールに供給するステップ(342)とをさらに含む上記(1)に記載の方法(300)。

(3) 前記許可を要求するステップ(311)が、前記基本モジュールの前記ローディングが完了するまで延期される上記(2)に記載の方法(300)。

(4) 前記授与された許可を改訂する前記ステップ(342~351)が、前記対応するライセンス情報を含む、各プログラムについてのデジタル・ライセンス証明書書を格納する別のコンピュータに、前記要求を示すメッセージを送信するステップ(343)と、前記対応するライセンス証明書で、前記要求の適合を検証するステップ(345~351)とを含む上記(2)または(3)に記載の方法(300)。

(5) 前記許可を授与する前記ステップが、前記コンピュータ上に格納され、前記対応するライセンス証明書の簡略化したコピーを含むローカル・デジタル・ライセンス証明書を検証するステップ(318~324)を含む上記(4)に記載の方法(300)。

(6) 前記ブートストラップの後に、前記別のコンピュータから受信するその新しいバージョンに従って各ローカル・ライセンス証明書を更新するステップ(354)とをさらに含む上記(5)に記載の方法(300)。

(7) 各プログラムの実行環境の変化を検出するために、前記コンピュータのオペレーションを監視するステップ(333)と、前記検出した変化に従って前記対応する授与された許可を改訂するステップ(336、360~384)とをさらに含む上記(1)ないし(6)に記載の方法(300)。

(8) コンピュータ上で動作するときに上記(1)ないし(7)のいずれかの前記方法を実行するための、前記コンピュータ(105)の作業メモリ(130c)中に直接ロード可能なコンピュータ・プログラム(230、235)。

(9) 上記(8)のプログラム(230、235)が格納されるコンピュータ可読記録媒体(150)。

(10) コンピュータ上のソフトウェア・プログラムの使用を制御する方法(300)であって、ライセンス交付管理システムに対して、各プログラムによって実行許可を要求するステップ(311; 327)と、前記プロ

グラムの使用の許可条件を示すライセンス交付情報の検証に従って前記許可を授与するステップ(318~321、342~351; 330、357~372)とを含み、前記コンピュータの実行環境での変化を検出するために、前記コンピュータのオペレーションを監視するステップ(333)と、前記検出した変化に従って前記授与された許可を改訂するステップ(336、360~384)とを特徴とする方法。

(11) コンピュータ上で動作するときに上記(10)の前記方法を実行するための、前記コンピュータ(105)の作業メモリ(130c)中に直接ロード可能なコンピュータ・プログラム(230、235)。

(12) 上記(11)のプログラム(230、235)が格納されるコンピュータ可読記録媒体(150)。

(13) ライセンス交付管理システム(230、235)と、コンピュータ(105)に対して基本サービスを提供する少なくとも1つのソフトウェア・プログラム(205c、215c)のローディングに關係するブートストラップを開始するための手段(135)とを含む前記コンピュータ上のソフトウェア・プログラムの使用を制御するためのシステム(100)において、各プログラムが、前記ライセンス交付管理システムに対して実行許可を要求するシステムであって、前記ライセンス交付システムが、前記ブートストラップの完了前に、プログラムの使用の許可条件を示すライセンス交付情報(240)の仮検証に従って前記許可を授与するための手段(230)と、前記ブートストラップの完了後に、ライセンス交付情報(225)の完全な検証に従って前記授与した許可を改訂するための手段(235)とを含むことを特徴とするシステム。

(14) ライセンス交付管理システム(230、235)を含むコンピュータ(105)上のソフトウェア・プログラム(205c、215c、250)の使用を制御するシステム(100)において、各プログラムが、前記ライセンス交付管理システムに対して実行許可を要求し、前記ライセンス交付管理システムが、ライセンス交付情報(225、240)の検証に従って許可を授与するシステムであって、前記プログラムの実行環境の変化を検出するために、コンピュータのオペレーションを監視するための手段(230、233)をさらに含む、前記ライセンス交付管理システムが、前記検出した変化に従って前記授与した許可を改訂することを特徴とするシステム。

【図面の簡単な説明】

【図1】本発明の方法を使用することができるネットワーク・システムの基本ブロック図である。

【図2】ネットワーク・システム中に含まれるクライアント・ワークステーションおよびサーバ・ワークステーションの作業メモリの部分的な内容である。

【図3】クライアント・ワークステーション上でソフト

ウェア・プログラムの使用を制御するための方法の流れ図である。

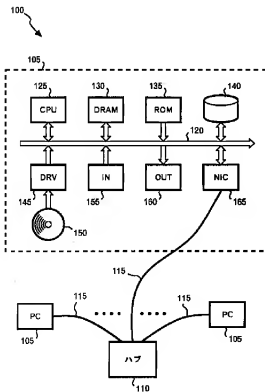
【図4】クライアント・ワークステーション上でソフトウェア・プログラムの使用を制御するための方法の流れ図である。

【図5】クライアント・ワークステーション上でソフトウェア・プログラムの使用を制御するための方法の流れ図である。

【符号の説明】

- 100 ネットワーキング・システム
- 105 ワークステーション
- 115 通信ケーブル
- 120 通信バス
- 125 中央演算処理装置 (CPU)
- 130 作業メモリ
- 135 読取り専用メモリ (ROM)
- 140 ハードディスク
- 145 ドライブ・ユニット (DRV)
- 150 C D-R O M
- 155 入力ユニット (I N)

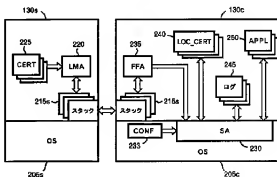
【図1】



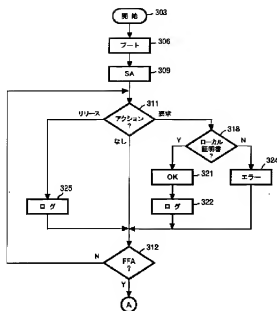
- 160 出力ユニット (O U T)
- 165 ネットワーク・インターフェース・カード (N I C)
- 130 c 作業メモリ
- 130 s 作業メモリ
- 205 c オペレーティング・システム (O S)
- 205 s オペレーティング・システム (O S)
- 215 c スタック
- 215 s スタック
- 220 ライセンス交付管理アプリケーション (L M A)
- 225 ライセンス証明書
- 230 ライセンス交付システム・エージェント (S A)
- 233 データ構造 (C O N F)
- 235 全機能ライセンス交付エージェント (F F A)
- 240 ローカル・ライセンス証明書
- 245 ログ構造 (L O G)
- 250 アプリケーション・プログラム

20

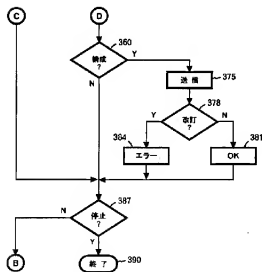
【図2】



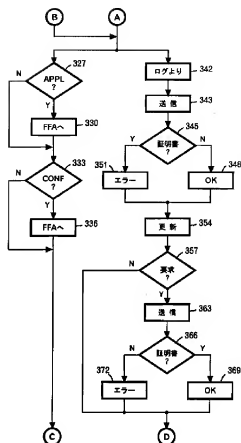
【図3】



【図5】



【図4】



フロントページの続き

(72)発明者 ジャンカルロ・カルボーネ
 イタリア ローマ 00142 ヴィア・ドウ
 ユッチャ・ディ・ブオニンセーニャ 29

(72)発明者 トーマス・フラン・シーレチ
 アメリカ合衆国07456 ニュー・ジャージ
 ー州リングウッド アルタ・ピスタ・ドラ
 イブ 58

Fターム(参考) 58076 B800 F801